ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 6, June 2014

Coding of video sequences using block matching motion estimation Three Step Search algorithms

Mr.S.M.Kulkarni[1] Dr.D.S.Bormane[2] Dr.S.L.Nalbalwar[3]

Research Scholar, JNTU Kakinada[1]

Principal, JSPM's Rajarshi Shahu College of Engg. , University of Pune, Pune[2]

Professor & Head (E&Tc), Dr. Babasaheb Ambedekar Technological University, Lonere, Raigad[3]

ABSTRACT: Video compression plays an important role in many digital video applications such as digital libraries, video on demand, and high definition television. Exploiting the temporal correlation between two successive sequence frames we can reduce enormously the memory space needed for compression. In this context, we have applied and compared different block matching motion estimation three step search algorithms. These algorithms include Three step search algorithm (TSS), New three steps search algorithm (NTSS), Improved TSS algorithm, Enhanced TSS algorithm, Fast TSS algorithm. The performance of an algorithm is a compromise between the peak signal to noise ratio and the computing search time of the algorithm. All the algorithms are compared on the basis of PSNR and number of searching points. Choosing an algorithm depends on what we require in application. Some applications need to be executed in a minimum of computing time(real time application) and others applications must give a good PSNR as it is in the medical image processing domain. Relative merits of all algorithms are given in conclusion.

KEY WORDS: Motion estimation, three step search, New three step search, block matching.

I. INTRODUCTION

Video compression plays an important role in many digital video applications such as digital libraries, video on demand, and high definition television. A video sequence with frame size of 176 × 144 pixels at 30 frames per second and 24 bits per pixel would require 18.25 Mbps, making it impractical to transmit the video sequence over standard telephone lines where data rates are typically restricted to 56,000 bits per second. This example illustrates the need for video compression. Effective video compression can be achieved by minimizing both spatial and temporal redundancy. A video sequence consists of a series of frames. In order to compress the video for efficient storage and transmission, the temporal redundancy among adjacent frames must be exploited. Temporal redundancy implies that adjacent frames are similar whereas spatial redundancy implies that neighboring pixels are similar. Video coding translates video sequences into an efficient bit stream. This translation involves the removal of redundant information from video sequence. Video sequence contains two kinds of redundancies spatial and temporal. Removal of spatial redundancy is generally termed as intraframe coding and removal of temporal redundancy is termed as interframe coding. The spatial redundancy within a frame is minimized by using transform coding. The commonly used transform is Discrete Cosine Transform. The temporal redundancy between successive frames is removed by interframe coding. Interframe coding relies on the fact that adjacent pictures in a video sequence have high temporal correlation. To minimize the temporal correlation, a frame is selected as a reference, and subsequent frames are predicted from the reference.

Motion estimation and compensation

Motion estimation describes the process of determining the motion between two or more frames in an image sequence. Motion compensation refers to the technique of predicting and reconstructing a frame using a given reference frame and a set of motion parameters. Motion compensation can be performed once an estimate of motion is available. Motion estimation and compensation have traditionally been performed using block-based methods [1]. They offer the advantage of being fast, easy to implement and fairly effective over a wide range of video content. Block-based motion estimation is the most practical approach to obtain motion compensated prediction frames. It divides frames into equally sized rectangular blocks and finds out the displacement of the best-matched block from previous frame as the motion vector to the block in the current frame within a search window. Based on block distortion measure or other matching criteria, the displacement of the best matched block will be described as the motion vector to the block in the current frame. The best match is evaluated by a cost function such as Mean Square Error (MSE), Mean Absolute Error (MAE), or Sum of Absolute Differences (SAD)[2].

This paper is organized as follows. In Section II, we present review of three step search technique. Section-III describes New three step search algorithm. Improved three-step search

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 6, June 2014

algorithm is explained in section-IV. Enhanced Three step search technique for motion estimation is elaborated in section-V. Fast three-step search algorithm is explained in section-VI. Simulation and comparison results are given in section-VII. Finally conclusion presented in section-VIII.

II. THREE STEP SEARCH ALGORITHM

Originally proposed by Koga et al. This is a fine-coarse search mechanism. This is fast algorithm for the reduction of computation. The general idea is represented in Fig 1. It starts with the search location at the center and sets the 'step size' S = 4, for a usual search parameter value of 7. It then searches at eight locations $\pm S$ pixels around location (0,0). From these nine locations searched so far it picks the one giving least cost and makes it the new search origin. It then sets the new step size S = S/2, and repeats similar search for two more iterations until S = 1. At that point it finds the location with the least cost function and the macro block at that location is the best match. The calculated motion vector is then saved for transmission [3].

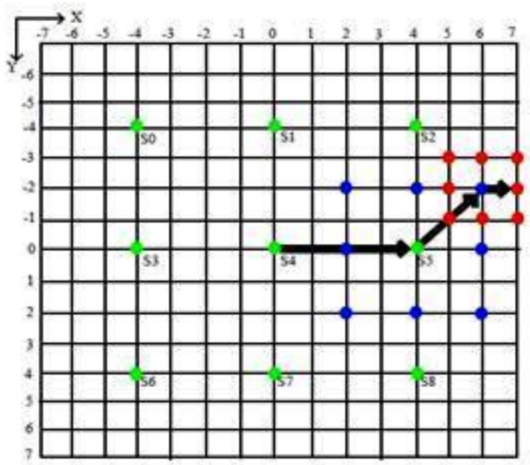


Fig.1. Three Step Search procedure

The number of candidate matching block processes for the TSS algorithm is remarkably reduced. TSS requires a fixed (9+8+8)=25 search points per block.

The matching of one macro block with another is based on the output of a cost function. The macro block that results in the least cost is the one that matches the closest to current block.

There are various cost functions, of which the most popular and less computationally expensive is Mean Absolute Difference (MAD) given by equation (1)

$$MAD \equiv \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|C_{ij} - R_{ij}\right| \qquad ---(1)$$

Another cost function is Mean Squared Error (MSE) given by equation (2)

$$MSE \equiv \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(C_{ij} - R_{ij})^2 \qquad ---(2)$$

Where N is the side of the macro bock, C_{ij} and R_{ij} are the pixels being compared in current macro block and reference macro block, respectively.

Peak-Signal-to-Noise-Ratio (PSNR) given by equation (3) characterizes the motion compensated image that is created by using motion vectors and macro clocks from the reference frame..

$$PSNR = 10\log_{10}\frac{(255)^2}{MSE} \qquad ---(3)$$

III. NEW THREE STEP SEARCH ALGORITHM

The new three step search algorithm (NTSS) has been proposed by Li, Zeng and Liou in 1994 . It is a modified version of the three step search algorithm for searching small motion in video sequences. For these video sequences, the motion vector distribution is highly center biased. Therefore, additional 8 neighboring checking points are searched in the first step of NTSS. Figure 2 shows two search paths with d=7. The center path shows the case of searching small motion. In this case, the minimum BDM point of the first step is one of the 8 neighboring checking points. . If the minimum BDM point is one of the eight points on the 3× 3 grid, additional three points (shown by squares)will be checked, The search is halfway stopped with matching three more neighboring checking points of the first step 's minimum BDM point. The number of checking points required is(17+3)=20. The upper right path shows the case of searching large motion. In this case, the minimum BDM point of the first step is one of the outer eight checking points [4]. Then the searching procedure proceeds in the same way as in the TSS algorithm. The number of checking points required in this step is (17+8+8)=33.

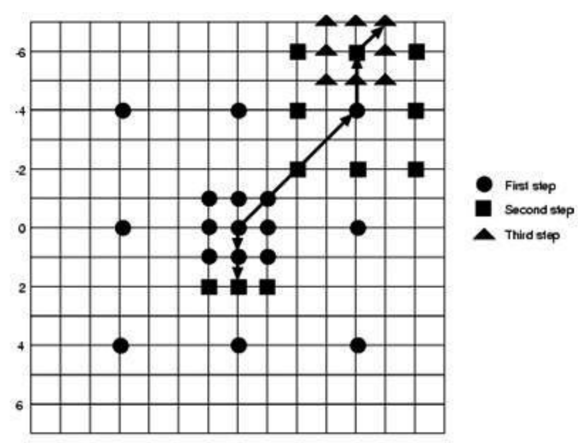


Fig. 2 New three step search algorithm.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 6, June 2014

IV. IMPROVED THREE STEP SEARCH ALGORITHM

TSS has unnecessarily many checking points when the motion is small for the block. Therefore a improved three-step search (ITSS) algorithm is proposed by Luo Tao et al. of Tianjin University, China. The algorithm was proposed to reduce more the computational complexity with the performance similar to TSS [5]. The ITSS will decrease more the number of the redundant checking points. The original three-step search is generally half-step search, that is to say, if the length of the first step is 4 (pixels), then the second step and the third step is 2 and 1 respectively. For the the Video system, the interval between two frames is very short (about 20 ms), which means that motion vector of the matching block between the two frames is relatively small. Based on this assumption, we adjust the length of three step in the algorithm from 4,2,1 to 3,2,1, then the search range is confined to ± 6.

As shown in Fig.3, The proposed ITSS algorithm utilizes a center-biased search pattern with nine checking points on a 7 × 7 grid in the first step. The SAD is evaluated for each of the 9 candidate search points. If the minimum SAD point is the search window center, the search will be terminated. Otherwise, the search window is reduced to 5 × 5 diamond Grid. Move the 5 × 5 diamond so that its center is the previous minimum SAD point. If the new SAD point obtained is located at the center position, search stop; otherwise, the search window is reduced to 3 × 3 diamond grid. In the same manner, compares SAD of 4 locations around the minimum SAD point of the second step and then gives the final motion vector.

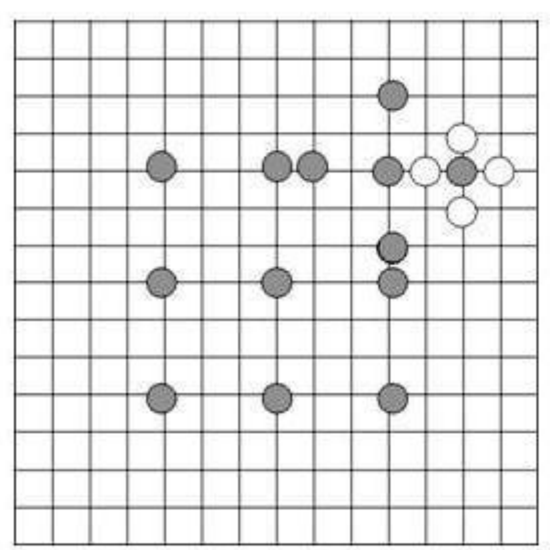


Fig.3 ITSS procedure

After adjustment, the accuracy of the motion estimation motion gets a certain improvement and compared to TSS, ITSS is much more robust, produces smaller motion compensation errors, and has a very compatible computational complexity.

V. ENHANCED THREE STEP SEARCH ALGORITHM

Prasad Lakamsani, Bing Zeng of Hong Kong University of Science and Technology have modified the first step of TSS in such a way that the concentration is more at the center: by intelligently arranging the available nine search points without increasing their number. Instead of spreading the search points all over the search window, they have tried to pull them as close to the center point as possible in the search window. In this way, they were able to reduce the distance between center point and surrounding points significantly. In other words, search pattern is more biased towards the center of the search window due to which the performance is expected to be far better than TSS[6].

With ETSS, as shown in figure 4, it is impossible to reach all of the 225 points in the search window of $15x15$ (i.e., -7 to +7). Even then, the amount of improvement is quite significant for H.261 applications where motion is small [7]. Performance improvement can be achieved further by decreasing the size of the search window to 11×11 (i.e., -5 to +5), instead of the normal search window of 15×15, in which case it is possible to reach any point in the search window. Their approach differs from TSS in the first step since it is very crucial and the most important step in hierarchical block matching methods. Once the best match is found in the first step, a local search will be carried out in second and third steps [8].

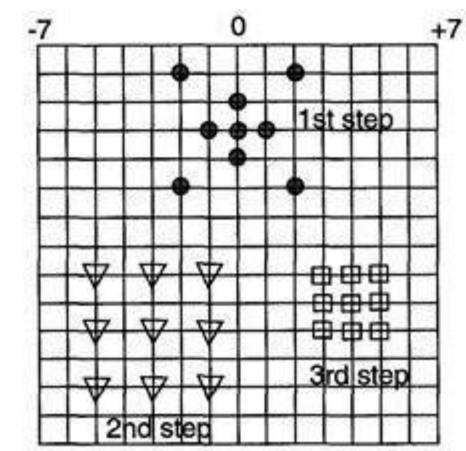


Fig. 4: three steps of ETSS

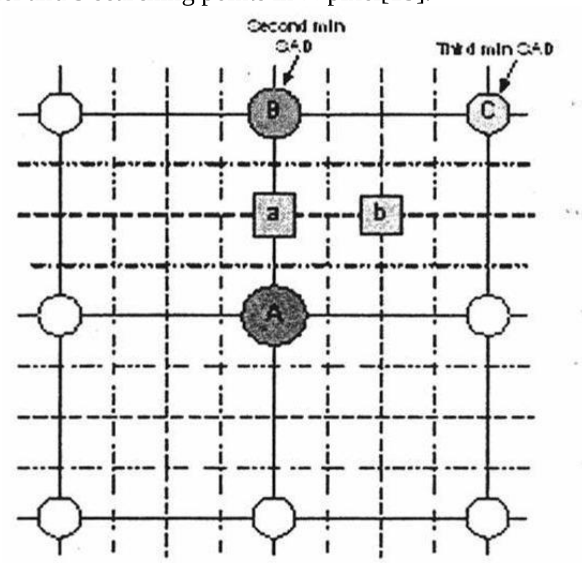
VI. FAST THREE STEP SEARCH ALGORITHM

Dongkyun Park et al. of University of Ulsan, Korea have proposed this method. They proposed ½ pixel and ¼ pixel Motion vectors which can be obtained through the position of integer pixel SAD value. If SAD value is minimum at(x, y) then ½ pixel motion vector can be obtained the following way [9].

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 6, June 2014

Step- 1: Find (x, y) with the smallest SAD in integer pixel and find the point which has the second smallest SAD and third smallest SAD in the neighboring integer pixel point within[10].
Step -2: Find ½ pixel SAD point between the second smallest SAD point and third smallest SAD Point .Considering that the ½ pixel searching point has the motion estimation in the direction from the minimum SAD to the second largest and the third large SAD, two searching points are enough[11]. Fig,5 shows ½ Pixel motion estimation.
In addition ¼ pixel searching point can be found using ½ pixel searching point, as shown in Fig.6. ¼ pixel SAD value can be found by using two points minimum integer SAD found in ½ pixel as in Fig 5[12].
Therefore, the strength of the proposed method is that the computation was made simpler and the Processing time was shortened because only 2 searching points were sought in ½ pixel and 3 searching points in ¼ pixel[13].



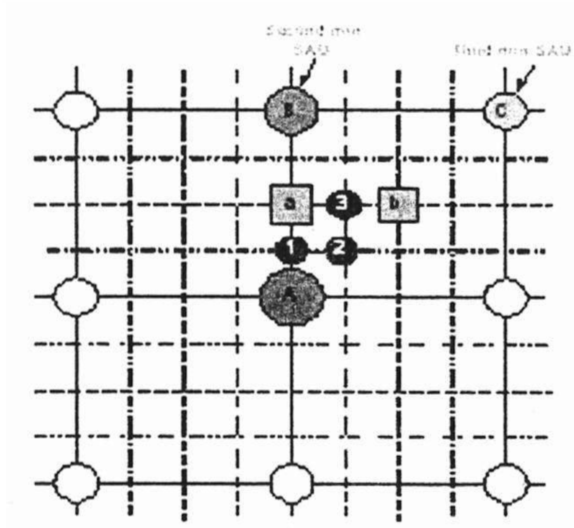


Fig.5. Proposed motion estimation in half pixel
Fig.6 Proposed motion estimation in quarter pixel

VII. SIMULATIONS AND COMPARISIONS

All the above algorithms are simulated using the luminance components of the first 150 frames of Claire, Miss America and Car phone video sequences. The image format for all sequences is QCIF (176 ×144). Each pixel is uniformly quantized to 8 bits. The block size is 16x16 pixels and the maximum motion displacement of search area is $\pm$ 7 in both horizontal and vertical directions. Minimum absolute difference (MAD) distortion function is used as the block distortion measure (BDM). The performance has been compared in terms of minimum square error (MSE) between estimated frames and original frames for the search algorithm. The computational speed up has been obtained by getting the average search number required for the motion vector estimation.
Table –I gives the experimental results for Average PSNR performance for all of above five techniques and Table-II gives the average search numbers for all of above five techniques.

Table-I AVERAGE PSNR ERFORMANCE

Sr. No.	Video sequence（100 frames）	PSNR				
		TSS	NTSS	Improved TSS	Enhan-ced TSS	Fast TSS
1	Foreman	42.35	44.34	40.07	43.01	32.23
2	Akiyo	34.37	36.56	32.56	35.57	24.45
3	Football	36.39	39.02	34.78	37.98	26.34

Table-II AVERAGE SEARCH NUMBERS

Sr.No	Name of technique	Avg. no. of searches
1	TSS	25
2	NTSS	(17+8+8)=33
3	Improved TSS	17
4	Enhanced TSS	27
5	Fast TSS	2+3=5

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 6, June 2014

VIII. CONCLUSION

ME is an important step in video image compression, it permits to reduce enormously the memory space needed for image compression. Choosing an algorithm depends on what we require in application. Some applications need to be executed in a minimum of computing time(real time application) and others applications must give a good PSNR as it is in the medical image processing domain.

Three Step search is one of the most popular BMAs and is also recommended by RM8 of H.261 and SM3 of MPEG owing to its simplicity and effectiveness. In the **New Three-Step Search Algorithm** the motion vector distribution is highly center biased. This algorithm gives good results when detecting small motion, but it is not suitable for large motion. **An improved three-step search algorithm** produces better quality performance and less computational time compared with three-step search (TSS) algorithm. This algorithm retains the simplicity and regularity of the original TSS method, works better than TSS in terms of motion compensation error and robustness, and is quite compatible to TSS in terms of computational complexity. **Enhanced Three Step Search Motion Estimation Algorithm** Studies show that the motion vector distribution within the search window is center-biased. This is used for video telephone and video conferencing applications where motion is very small. Though the number of search points is the same as in TSS, far better performance has been achieved with this approach. The probability of the minimum SAD values is high when searching point is in the distance one pixel from the reference point. Thus by reducing the searching area **Fast Three Step Search Motion Estimation Algorithm** had overcome the computational complexity problem. The time that it took to calculate was reduced by approximately 24% for encoder processing and by approximately 51% for sub pixel motion estimation

REFERENCES

[1] K.Laidi, M.A.Bailiche, M.Mehenni," Comparative Study of Block Matching Techniques Used in Video Image Motions Estimation", Proceedings of the 5th International Symposium on image and Signal Processing and Analysis 2007.
[2] Fulvio Moschetti Murat Kunt and Eric Debes, "A statical adaptive block matching motion estimation", IEEE transactions on circuits and systems for video technology, Vol 13 No. 4APril 2003.
[3] Kuo-Liang Chung and Lung-Chun Chang, "A new predictive search area approach for fast block motion estimation", IEEE transactions on image processing, Vol 12.No 6, Jun 2003.
[4] R. Li et *al.*, "A new three-step search algorithm for block motion estimation", IEEE Trans. on Circuits and Systems for Video Technology, Aug. 1994.
[5] Luo Tao, Yao Su-ying, Shi Zai-feng, Gao peng," A improved three-step search algorithm with Zero Detection and vector filter for motion estimation", 2008 International Conference on Computer Science and Software Engineering 978-0-7695-3336-0/08
[6] Xuan Jing and Lap-Pui Chau, "An Efficient Three-Step Search Algorithm For Block Motion Estimation", IEEE transactions on multimedia, Vol 6, No 3, Jun 2004.
[7] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block matching motion estimation," IEEE Trans. Image Processing, Vol. 9, No. 2, pp. 287-290, Feb. 2000.
[8] Prasad Lakamsani, Bing Zeng, and Ming Liou," An Enhanced Three Step Search Motion Estimation Method", 0-IEEE 7803-3073-0/96/1996 pp 754-757
[9] Jinwen Zan, M. Omair Ahmed, M. N. S. Swamy, "A Multiresolution motion estimation technique with indexing", IEEE transactions on circuits and systems for video technology, Vol 16, No.. 2, February 2006.
[10] Cheng Du, Yun He, and Junli Zheng, PPHS: A Parabolic Prediction-Based, Fast Half-Pixel Search Algorithm for Very Low Bit-Rate Moving-Picture Coding", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No.6, June 2003.
[11] H. M. Wong, O. C. Au, Huang, S. Zhang and W. Z. Yan, "Sub..optimal Quarter- Pixel Inter-Prediction Algorithm (SQIA)," ICASSP '05. IEEE International Conference on Volume 2, pp. 921-924, March 18-23,2005.
[12] Chung-Neng Wang, Shin-Wei Yang, Chi-Min Liu and Tihao Chiang, "A hierarchical decimation lattice based on Nqueen with an application for motion estimation" , IEEE signal processing letters, Vol. 10, No. 8, Aug.2003.
[13] Atsuro Ichigaya, Masaai Kurozumi, Naohiro Hara, Yukihiro Nishida and Eisuke Nakasu, "A Method of Estimating Coding PSNR Using Quantized DCT Coefficients", IEEE transactions on circuits and systems for video technology, Vol. 16, No. 2, February 2006.